SHValidateUNC

Vulnerable to TOCTOU issues

Sean Barnum, Cigital, Inc. [vita[1]]

Copyright © 2007 Cigital, Inc.

2007-04-16

Part "Original Cigital Coding Rule in XML"

Mime-type: text/xml, size: 5716 bytes

Attack Category	• Path spoofing or confusion problem		
Vulnerability Category	• Indeterminate File/Path • TOCTOU - Time of Check, Time of Use		
Software Context	• File Path Management		
Location	• shlobj.h		
Description	Validates a Universal Naming Convention (UNC) path by calling WNetAddConnection3. The function makes it possible for the user to type a remote network access (RNA) UNC application or document name from the Run dialog box on the Start menu. SHValidateUNC is vulnerable to TOCTOU attacks.		
APIs	**Function Name**	**Comments**	
	SHValidateUNC	check	
Method of Attack	The key issue with respect to TOCTOU vulnerabilities is that programs make assumptions about atomicity of actions. It is assumed that checking the state or identity of a targeted resource followed by an action on that resource is all one action. In reality, there is a period of time between the check and the use that allows either an attacker to intentionally or another interleaved process or thread to unintentionally change the state of the targeted resource and yield unexpected and undesired results.		
Exception Criteria			
Solutions	**Solution Applicability**	**Solution Description**	**Solution Efficacy**
	Generally applicable.	Utilize a file descriptor version of stat/ fstat when checking.	Efficient

---

1.   http://buildsecurityin.us-cert.gov/bsi-rules/35-BSI.html (Barnum, Sean)

Generally applicable.	The most basic advice for TOCTOU vulnerabilities is to not perform a check before the use. This does not resolve the underlying issue of the execution of a function on a resource whose state and identity cannot be assured, but it does help to limit the false sense of security given by the check. Attempt to create the directory and then check status after the creation.	Does not resolve the underlying vulnerability but limits the false sense of security given by the check.
Generally applicable.	Limit the interleaving of operations on files from multiple processes.	Does not eliminate the underlying vulnerability but can help make it more difficult to exploit.
Generally applicable.	Limit the spread of time (cycles) between the check and use of a resource.	Does not eliminate the underlying vulnerability but can help make it more difficult to exploit.
Generally applicable.	Recheck the resource after the use call to verify that the action was taken appropriately.	Effective in some cases.

Signature Details	BOOL SHValidateUNC(HWND hwndOwner,LPWSTR pszFile,UINT fConnect);
| Examples of Incorrect Code | ```
int use_status;
struct stat statbuf;
check_status=stat("pszfile",
&statbuf);
[...]
int ret = SHValidateUNC(hwndOwner,
"pszFile", fConnect);
[...]
``` |
| Examples of Corrected Code | ```
[...]
int ret = SHValidateUNC(hwndOwner,
"pszFile", fConnect);
[...]
``` |
| Source Reference | • http://msdn.microsoft.com/library/default.asp?url=/library/en-us/shellcc/platform/shell/reference/functions/shvalidateunc.asp[2] |
| Recommended Resource | |

Discriminant Set	Operating System	• Windows
	Languages	• C
		• C++

Cigital, Inc. Copyright

1. mailto:copyright@cigital.com